In the Specification

Please substitute the following amended paragraph for the first paragraph on page 7, lines 1-18, as follows:

FIG. 4 illustrates the printing workflow system 2 interacting with a cell in a network provided in a print server. The product cell controller 34 for the cell receives a sub-job 48 from the server 20 to be further processed by the cell. The server 20 stores in its storage 23 the capacities and capabilities of each cell in the print shop to produce different product-types. For example, cell 32 in the network produces three different types of documents and cell 40 produces two types of documents. (It is quite possible that two different cells can produce similar, or even the same, document types. A document type is uniquely characterized by the sequence of processing steps to completely finish the document). The server 20 stores this information to determine which cell has the capabilities to process a document job. The printing workflow system 23 also stores the capacity of each cell to determine the volume of a particular product-type that a cell can produce. As stated above, the job decomposition module 14 splits a document processing job into sub-jobs to be sent to various autonomous cells for processing. The cells in the network are autonomous and can produce their respective product entirely by themselves. Thus, in the example shown in Fig. 4, a document processing job is split into sub-jobs 48 and 50 that are sent to cells 32 and 40, respectively. The product cell controllers 34 and 42 send the sub-jobs 48 and 50 to devices 36a, 36b, 36c and 44a, 44b, 44c in the respective cells 32 and 40 for processing.

Please substitute the following amended paragraph for the second paragraph on page 7, lines 20-26, as follows:

FIG. 5 illustrates an example of how capacity is defined for a cell in the illustrative embodiment. As stated above, the printing workflow system 2 stores the capacity of each cell. "Capacity" is the maximum volume of a particular product type that the cell can produce for a time period. For example, **FIG. 5** shows capacities for 31, 33, and 35 three different product types (Product A, Product B, and Product C). The printing workflow system 2 updates the capacities and makes it easier to determine which cells should be assigned a sub-job. Capabilities are used to determine the assignment for a cell to process a sub-job.

Please substitute the following amended paragraph for the first full paragraph on page 8, lines 6-11, as follows:

A3

FIG. 7 illustrates a flow diagram of the inventive job scheduling system. Once a document processing job is prepared for processing, the job scheduling system 46 determines the best method for scheduling the document processing job. The present invention provides two techniques for routing and scheduling document processing jobs. The first technique 47a is the market-based approach of scheduling a document processing job. The second technique is an optimization method.

Please substitute the following amended paragraph for the second full paragraph on page 8, lines 13-21, as follows:

A4

The first technique 47a for scheduling a document processing job is based on an arrangement wherein a document factory have autonomous cells that do not necessarily share their internal operational characteristics with each other. Operational characteristics such as cell capacities, cell capabilities, and slack times are not necessarily maintained globally by the printing workflow system 2. As further discussed below, the printing workflow system 2 sends a description of a document processing job to each autonomous cell in the printing workflow system. Each cell responds to the printing workflow system with a bid to process the document processing job. Based on the bid, the printing workflow system decides which cell or cells will process this document processing job.

Please substitute the following amended paragraph for the third full paragraph on page 8, lines 23-29, as follows:

A5

The second technique 47b for scheduling a document processing job is based on the optimization of the overall global arrangement of the cells in the printing workflow system 2. As further discussed below, several of the operating characteristics of each of the cells in the printing workflow system can be optimized. The optimization of the operating characteristics is based on finding the Pareto optimal solutions for scheduling the document processing job. Using this technique requires that operating characteristics of all the cells be maintained globally in the printing workflow system.

Please substitute the following amended paragraph for the third full paragraph on page 9, lines 16-24, as follows:

A6

As demonstrated on step 51, the vector **O** is ordered in a fixed sequence. This sequence is fixed for a given printing workflow system. If the printing workflow system gets augmented with newer operations, then the new operations are prepended to the matrix, i.e. they are added to the beginning of the vector. For each document processing job, the operations required to complete the job are determined. For each operation that needs to be completed a numerical value of "1" is assigned for the job, and a numerical value of "0" is assigned if the operation is not needed, as shown in step 52. This will result in a new vector, which resembles a binary string. The binary string is converted to its decimal equivalent, as shown in step 54. That represents the unique ID for the document processing job.

Please substitute the following amended paragraph for the first full paragraph on page 10, lines 1-22, as follows:

A7

For a given document processing job and a given integer number D the steps for assigning a descriptive ID are as follows. In step 56, the printing workflow system computes a unique ID (as disclosed in FIG. 7) for the document processing job. The unique ID is used as a basis to develop a descriptive ID for the document processing job. Step 58 discloses appending the due date to the unique ID using D-digit month-day format (e.g., 0623 for June

23 where D = 4). The due time is appended to the unique ID in step 60 using a D-digit 24-hour format (e.g. 1525 for 3:25 PM where D = 4. Steps 58 and 60 allow the printing workflow system 2 or a cell 4, 6 or 8 access to the due date and the due time of a job by looking at the descriptive ID of the document processing job without any exhaustive computation. The number of duplicates of the originals that are needed are appended to the unique ID using a D-digit format (step 62). If this value exceeds D-digits, the document processing job is partitioned into subjobs such that their number of duplicates fit into D-digits. Step 64 discloses appending the unique ID to the number of units (in the original) associated to the operation O1 only if this operation is required (e.g., if O1 denotes color printing and the original has 26 colors copies and D=4, append 0026). The last step is repeated for the next operation on the sequence **O**, and so on and so forth until all operations **O** are considered. Step 66 entails combining all of the aforementioned appended entities into a string. This results in a decimal string with (at least five) different D-digit fields. Conversion to hexadecimal can be used to save some digits; i.e., generate the descriptive ID in decimal, convert to hexadecimal for ID transmission, etc., and convert to back to decimal for ID interpretation.

A7 cont

Please substitute the following amended paragraph for the first full paragraph on page 11, lines 9-18, as follows:

A8

The job-scheduling problem may be stated more precisely. Suppose that a LDF has m manufacturing cells, C1 (72), C2 (74), ..., Cm (76), and n jobs, J1 (67), J2 (68), ..., Jn (70), waiting to be processed. Each one of the jobs has a customer due date (the job has to be finished by this date). Decide what portions of each one of the n jobs are to be assigned to each one of the m cells and in which order these job portions are to be queued in a given cell, such that all jobs are finished by their customer due dates 78, 80, 82 and some additional objective is achieved. Here, "to finish all jobs by their customer due dates" is the hard constraint. A candidate schedule not meeting this constraint is discarded; otherwise it is a *feasible* schedule. The "additional objective" is the soft goal; e.g., to make each cell's processing time as small as possible.

Please substitute the following amended paragraph for the third paragraph on page 15, lines 20-31, and extending onto page 16, lines 1-3, as follows:

A9

FIG. 13 illustrates an example of bidding among cells in the document processing jobs in the printing workflow system. The printing workflow system consists of autonomous cells that can process most of the jobs all within themselves as they arrive in the shop. Each cell can autonomously manufacture a finite number of products. When document jobs arrive in the printing workflow system, it is crucial to manage the flow of these jobs through the different cells especially when a given job can be manufactured in more than one cell. Jobs arrive at the LPPS in step 110, and they are pooled into the LPPS in step 112. Select one job from this pool in 114, and in 116 check if a single cell can complete the job. If the answer is "no", the job is partitioned in 118 into sub-jobs that can be finished in a single cell; and later in 120 each subjob is mapped into a new job and added to the server pool. If the answer is "yes", determine which cells can do the job in 122 and obtain job bids from these cells in 124. In step 126 we remove the job from the LPPS pool and send it to the cell issuing the lowest bid. In 128 we check if the LPPS pool is empty. If it is empty, we wait (130) for some period of time (say, 5 minutes) and check the LPPS pool again. If there are jobs in the pool, we go back to 114 and select a job from the LPPS pool.

Please substitute the following amended paragraph for the second full paragraph on page 16, lines 13-31, as follows:

A10

FIG. 14 illustrates one example on how cells submit their bids to process a document job. Once cell 160 receives information from the printing workflow system regarding a document processing job it prepares to submit a bid to the printing workflow system by doing several step as shown in FIG. 14. When a job arrives to the cell (step 160) the process routings (i.e. the sequence of operations in the cell needed to fully execute the job) needed to complete the job are determined in step 162. The cell determines the required processing time for the complete document processing job for each of these routings using the optimal batch size as depicted in step 164. The optimal batch size is desired because it allows for the efficient processing of the document processing job. The cell sorts all the jobs by their priorities such that the one with the highest priority is the first one to be released for production. The cell determines the estimated release time for production of the job as

depicted in step 168. Step 170 determines if the job can be fully completed in the current cell, determine the slack time S for the current job where the slack time is defines as (Due Date – Release Time – Estimated Production Time), and define a decreasing cost function as F(S) = JobSize/S as shown in step 174 to generate a bid (step 176). Otherwise, if the document processing job is large enough not to be completed fully by a cell, then the cell 160 will inform the LPPS about the maximum fraction of the job it can process (step 175 and also provide the LPPS with a monotonically increasing cost function of the job fraction (step 177). The LPPS will evaluate all the capable cells to see if the job can be fully completed in neither.

A10 cont.

Please substitute the following amended paragraph for the first full paragraph on page 17, lines 1-13, as follows:

A11

FIG.15 illustrates in a flowchart the steps needed to determine a bid for a document-processing job that is large enough not to be fully completed by any cell (step 132). The printing workflow system determines what fraction of the entire job can be completed within the different capable cells as shown in step 134. Let us assume that we have k cells that are capable of doing the job and are bidding for portions of the job and that each can perform a maximum of $\alpha jmax$ where $\{1 <= j <= k\}$ fraction of the job. Each cell provides a monotonically increasing cost function of α for this document-processing job denoted by $Cj(\alpha j)$ as shown in step 136. For example, for cell j the cost function could be $Cj(\alpha j) = 7\,\alpha j + 10$. The server optimizes the total cost function $C(\alpha 1,\ldots, \alpha k) = \sum_{j=1}^{k} Cj(\alpha j)$ with constraints ($\alpha j < \alpha jmax$, $\alpha 1+\alpha 2+\ldots+\alpha k=1$) to determine the fraction of jobs that will go to each cell. The cost functions provided by the cells can be different for each cell. The server then creates sub jobs based on the optimal fractions and routes them to appropriate cells (step 140).